Policy Generation with Probabilistic Guarantees for Long-term Autonomy of a Mobile Robot

Bruno Lacerda,* David Parker† and Nick Hawes*

Overview. In this demo, we will illustrate our work on the use of formal verification techniques, in particular probabilistic model checking, to enable long term deployments of mobile service robots in everyday environments. Our framework generates policies to meet specifications given in the co-safe fragment of linear temporal logic (LTL) over Markov decision process (MDP) models of mobile robot behaviour. For a given specification over these models, our framework produces a policy that maximises the probability of the robot satisfying the specification, whilst minimising the expected time to do so. The policy obtained for this objective can be seen as a robot plan with attached probabilistic performance guarantees.

Environmental Modelling. Our models for planning and execution of mobile robot tasks are based on a discretisation of the environment based on a *topological map* (see Fig 1). As it navigates, the robot gathers data on the failure or success of navigation through an edge, and the time taken to do so. We use this data to build probabilistic edge models, for example using the technique presented in [1]. This, along with the specification of other actions that can be active in different locations, allows us to build MDP models of robot navigation and action execution where the probabilistic outcomes are learnt from experience.

Partial Satisfaction of Co-Safe LTL Specifications. In order to generate policies for these models, we use our work [2] on cost-optimal policy generation for co-safe LTL specifications that are not satisfiable with probability one. The overall objective of the work is to generate policies that maximise the *probability of success* and minimise the *undiscounted expected cumulative cost* to achieve the co-safe LTL task. Furthermore, we tackle the question of what to do when the task becomes unsatisfiable during execution. In many cases, even if the probability of satisfying the overall task is zero, it is still possible to fulfil part of it. An illustrative example is a robot that needs to navigate to every office in a building to perform security checks. During execution some doors might be closed, making offices inaccessible. This will make the overall task unsatisfiable, yet we still want the robot to check as many offices as it can. We formalise this notion as a progression reward defined over the automaton for the LTL specification.

Given an MDP and a co-safe LTL specification, we show that the problems of (i) maximising the probability of satisfying a co-safe LTL formula; (ii) maximising the

*Oxford Robotics Institute, University of Oxford, UK.
†School of Computer Science, University of Birmingham, UK.

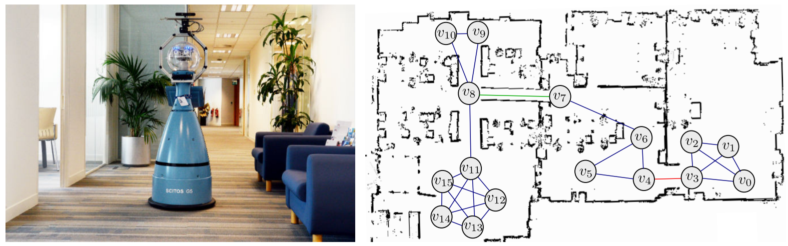

Figure 1: The mobile service robot we aim to use for the demo, deployed in an office scenario (left) and a topological map (right). Edges in blue represent the default laser based navigation, the edge in green represents a specialised ramp traversal behaviour, and the edge in red represents a specialised door crossing behaviour.

progression reward (i.e., fulfilling as much of the specification as possible); and (iii) minimising a cost function while performing (i) and (ii) can be solved independently by standard techniques in a *pruned product MDP*. Furthermore, in [3], we introduced a *nested* version of value iteration that allow us to implement prioritisation of the three objectives above in an efficient manner.

Implementation. Our policy generation approach is implemented within the PRISM probabilistic model checker[1] and is fully integrated within the Robot Operating System (ROS) middleware[2], and has been used extensively for deployments of mobile service robots for long periods of time [4]. Fig. 1 depicts our robot in one such deployment.

Demo Proposal. Our proposal is to showcase this framework live during the workshop, deploying our robot in the workshop venue and having it perform tasks throughout the day (the robot is based in the Oxford Robotics Institute, hence it can be easily moved to the workshop venue). In conjunction with showing the live robot behaviour, we will, among other things, provide visualisation of the generated policies on a map of the environment; showcase how the robot keeps track of the performance guarantees calculated during policy generation while it is executing; and show how these guarantees can be used for execution monitoring[3].

References

[1] J. Pulido Fentanes, B. Lacerda, T. Krajník, N. Hawes, and M. Hanheide, "Now or later? predicting and maximising success of navigation actions from long-term experience," in *ICRA*, 2015.

[2] B. Lacerda, D. Parker, and N. Hawes, "Optimal policy generation for partially satisfiable co-safe LTL specifications," in *IJCAI*, 2015.

[3] B. Lacerda, D. Parker, and N. Hawes, "Nested value iteration for partially satisfiable co-safe LTL specifications," in *AAAI Fall Symposium on Sequential Decision Making for Intelligent Agents*, 2015.

[4] N. Hawes et al, "The STRANDS project: Long-term autonomy in everyday environments," *IEEE Robotics and Automation Magazine*, vol. 24, no. 3, pp. 146–156, 2017.

[1] http://www.prismmodelchecker.org/

[2] http://www.ros.org/

[3] See https://youtu.be/SBslh6OPq9U for an example of the robot executing a policy.